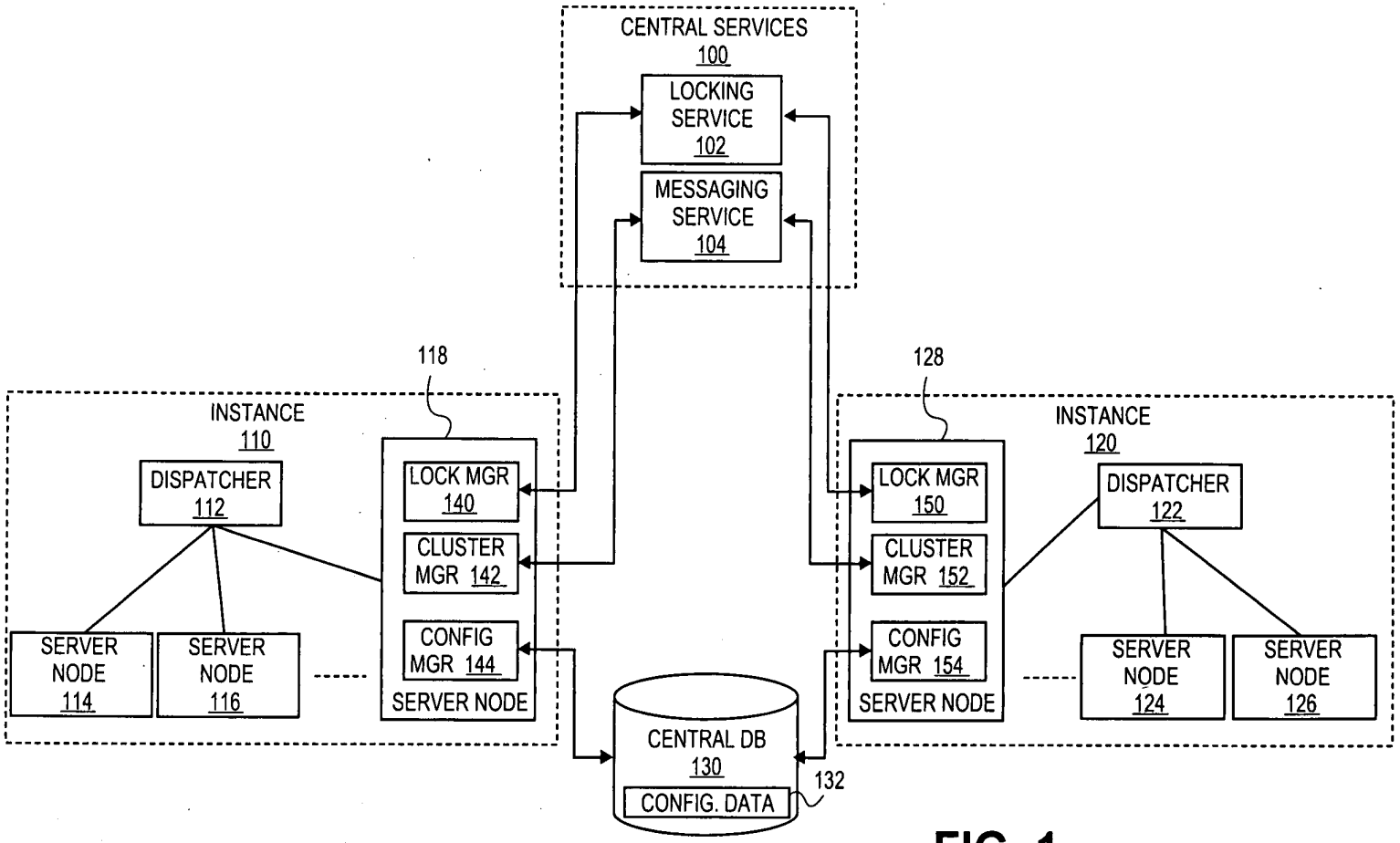
CENTRAL SERVICES
100
LOCKING SERVICE
102
MESSAGING SERVICE
104
118
128
INSTANCE
110
INSTANCE
120
DISPATCHER
112
LOCK MGR
140
CLUSTER MGR 142
CONFIG MGR 144
SERVER NODE
LOCK MGR
150
CLUSTER MGR 152
CONFIG MGR 154
SERVER NODE
DISPATCHER
122
SERVER NODE
114
SERVER NODE
116
SERVER NODE
124
SERVER NODE
126
CENTRAL DB
130
CONFIG. DATA
132

FIG. 1

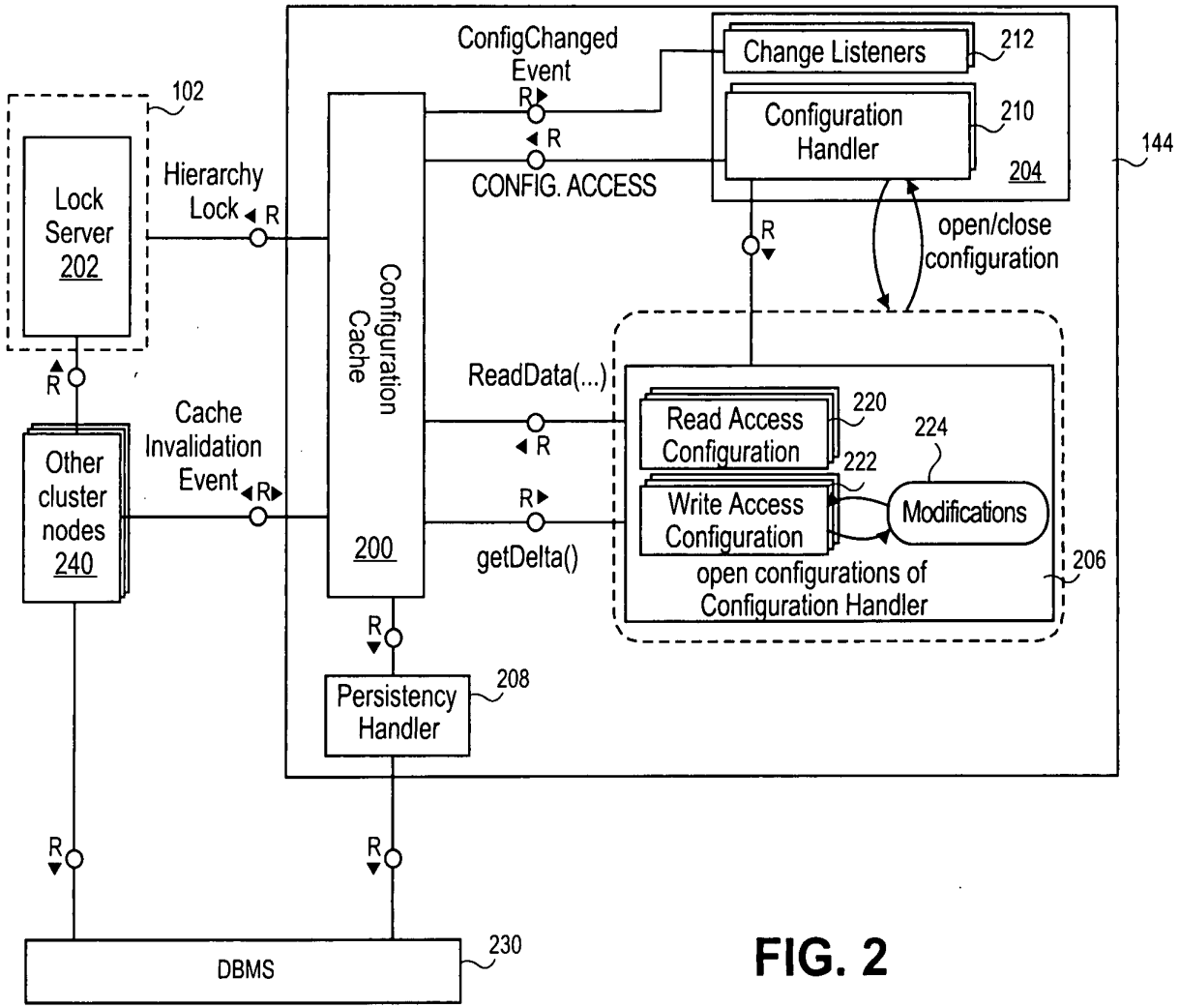
ConfigChanged Event
Change Listeners
212
Configuration Handler
210
204
144
102
Lock Server 202
Hierarchy Lock
CONFIG. ACCESS
open/close configuration
Configuration Cache
200
ReadData(...)
Read Access Configuration
220
224
222
Write Access Configuration
Modifications
Other cluster nodes 240
Cache Invalidation Event
getDelta()
open configurations of Configuration Handler
206
Persistency Handler
208
DBMS
230
R

FIG. 2

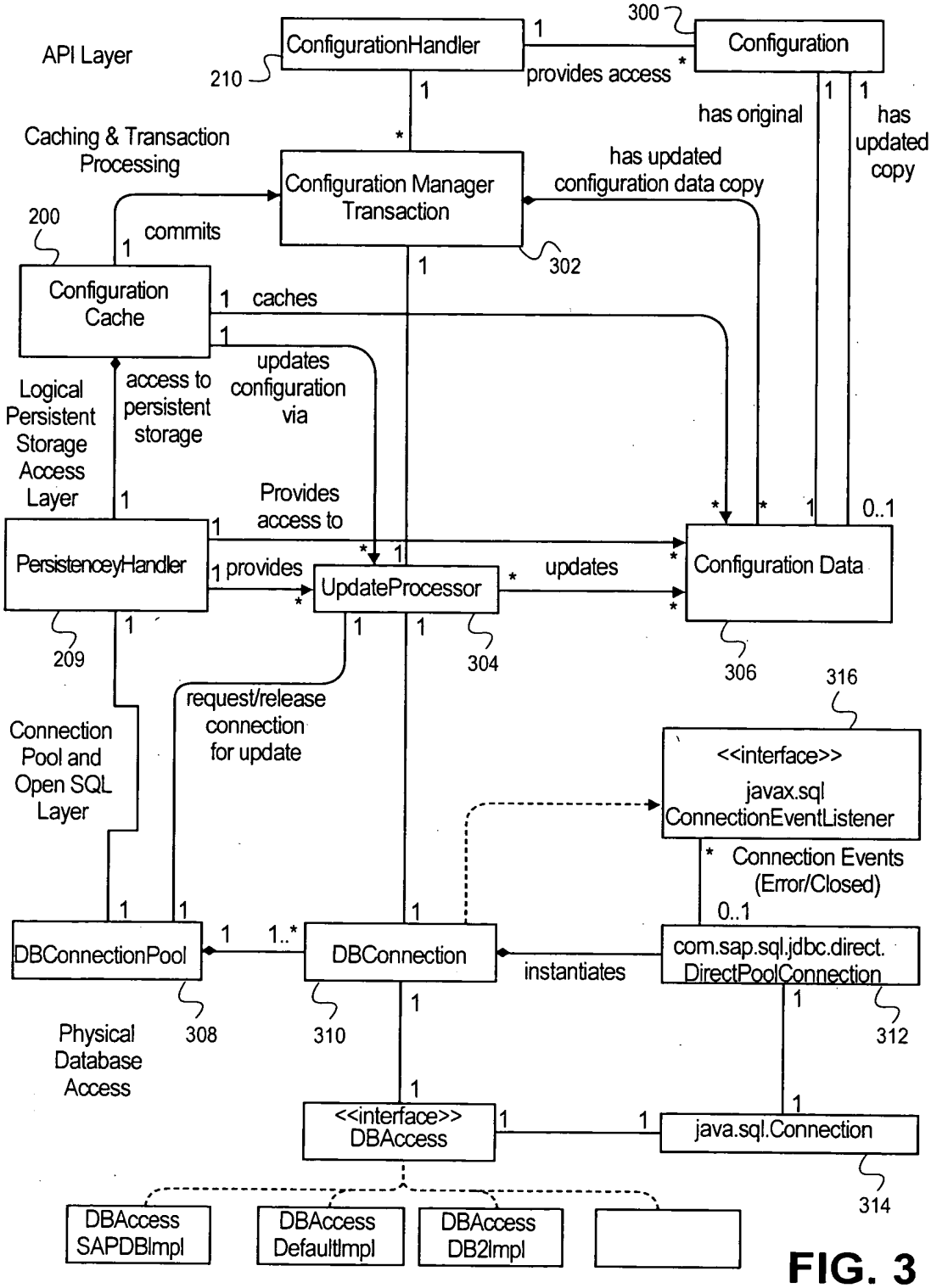
API Layer
ConfigurationHandler
210
300
Configuration
provides access
has original
has updated copy
Caching & Transaction Processing
Configuration Manager Transaction
has updated configuration data copy
200
commits
302
Configuration Cache
caches
updates configuration via
Logical Persistent Storage Access Layer
access to persistent storage
Provides access to
PersistenceyHandler
provides
UpdateProcessor
updates
Configuration Data
0..1
209
304
306
316
request/release connection for update
Connection Pool and Open SQL Layer
<<interface>> javax.sql ConnectionEventListener
Connection Events (Error/Closed)
DBConnectionPool
DBConnection
instantiates
com.sap.sql.jdbc.direct. DirectPoolConnection
Physical Database Access
308
310
312
<<interface>> DBAccess
java.sql.Connection
314
DBAccess SAPDBImpl
DBAccess DefaultImpl
DBAccess DB2Impl

FIG. 3

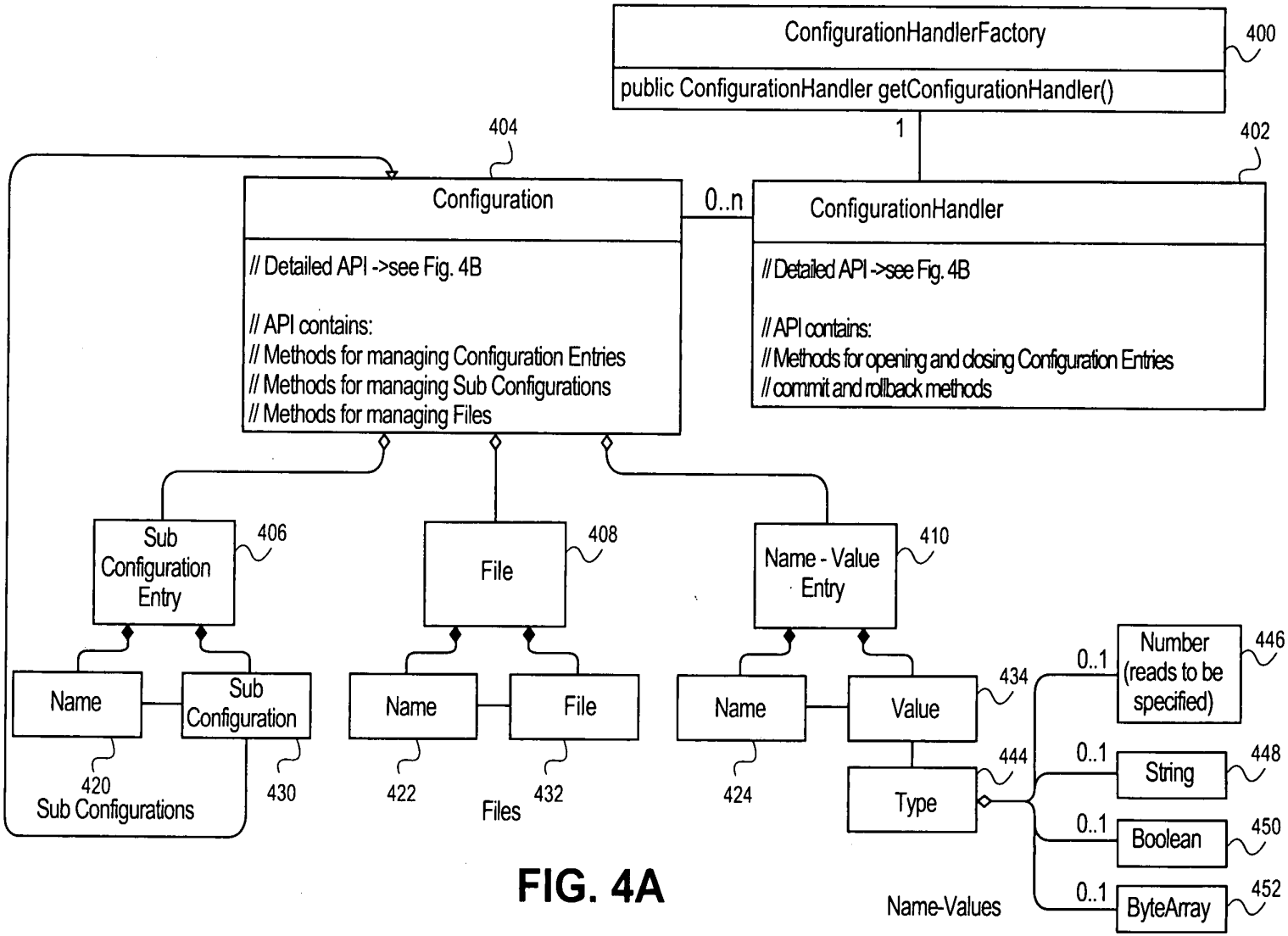
ConfigurationHandlerFactory
400
public ConfigurationHandler getConfigurationHandler()
1
402
404
Configuration
0..n
ConfigurationHandler
// Detailed API ->see Fig. 4B
// API contains:
// Methods for managing Configuration Entries
// Methods for managing Sub Configurations
// Methods for managing Files
// Detailed API ->see Fig. 4B
// API contains:
// Methods for opening and closing Configuration Entries
// commit and rollback methods
Sub Configuration Entry
406
File
408
Name - Value Entry
410
Name
Sub Configuration
Name
File
Name
Value
434
0..1
Number (reads to be specified)
446
444
0..1
String
448
Type
0..1
Boolean
450
0..1
ByteArray
452
420
Sub Configurations
430
422
Files
432
424
Name-Values

FIG. 4A

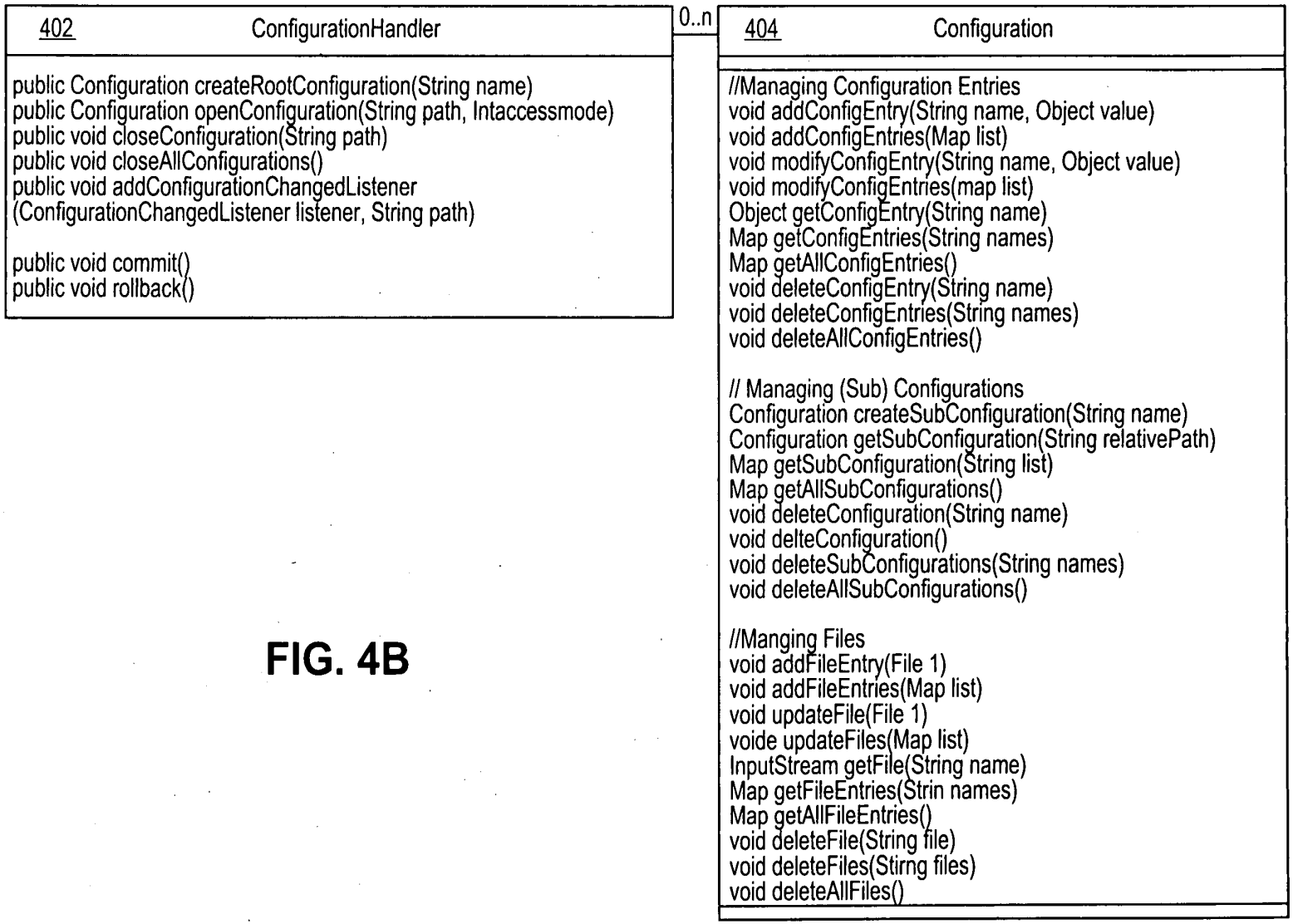
402
ConfigurationHandler
public Configuration createRootConfiguration(String name)
public Configuration openConfiguration(String path, Intaccessmode)
public void closeConfiguration(String path)
public void closeAllConfigurations()
public void addConfigurationChangedListener
(ConfigurationChangedListener listener, String path)
public void commit()
public void rollback()
0..n
404
Configuration
//Managing Configuration Entries
void addConfigEntry(String name, Object value)
void addConfigEntries(Map list)
void modifyConfigEntry(String name, Object value)
void modifyConfigEntries(map list)
Object getConfigEntry(String name)
Map getConfigEntries(String names)
Map getAllConfigEntries()
void deleteConfigEntry(String name)
void deleteConfigEntries(String names)
void deleteAllConfigEntries()
// Managing (Sub) Configurations
Configuration createSubConfiguration(String name)
Configuration getSubConfiguration(String relativePath)
Map getSubConfiguration(String list)
Map getAllSubConfigurations()
void deleteConfiguration(String name)
void delteConfiguration()
void deleteSubConfigurations(String names)
void deleteAllSubConfigurations()
//Manging Files
void addFileEntry(File 1)
void addFileEntries(Map list)
void updateFile(File 1)
voide updateFiles(Map list)
InputStream getFile(String name)
Map getFileEntries(Strin names)
Map getAllFileEntries()
void deleteFile(String file)
void deleteFiles(Stirng files)
void deleteAllFiles()

FIG. 4B

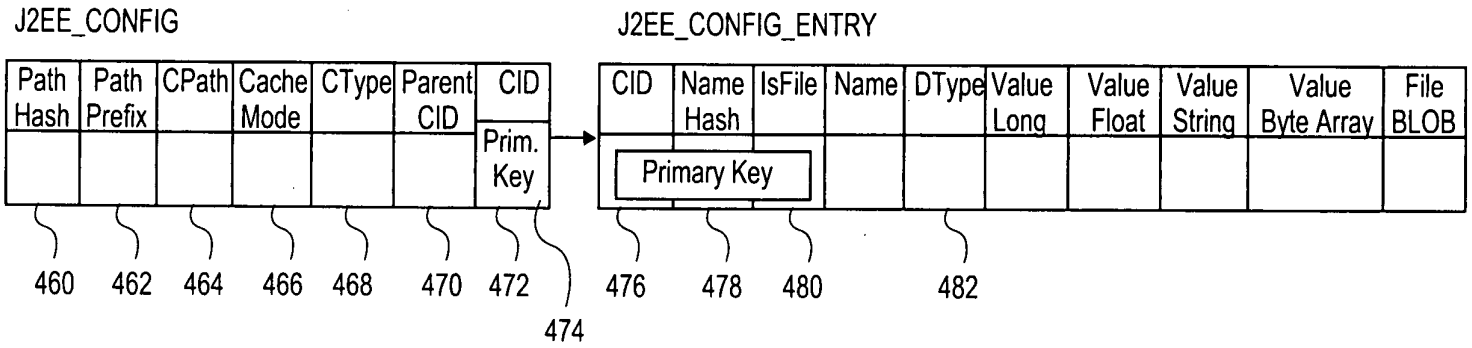
J2EE_CONFIG
Path Hash
Path Prefix
CPath
Cache Mode
CType
Parent CID
CID
Prim. Key
J2EE_CONFIG_ENTRY
CID
Name Hash
IsFile
Name
DType
Value Long
Value Float
Value String
Value Byte Array
File BLOB
Primary Key
460
462
464
466
468
470
472
474
476
478
480
482

FIG. 4C

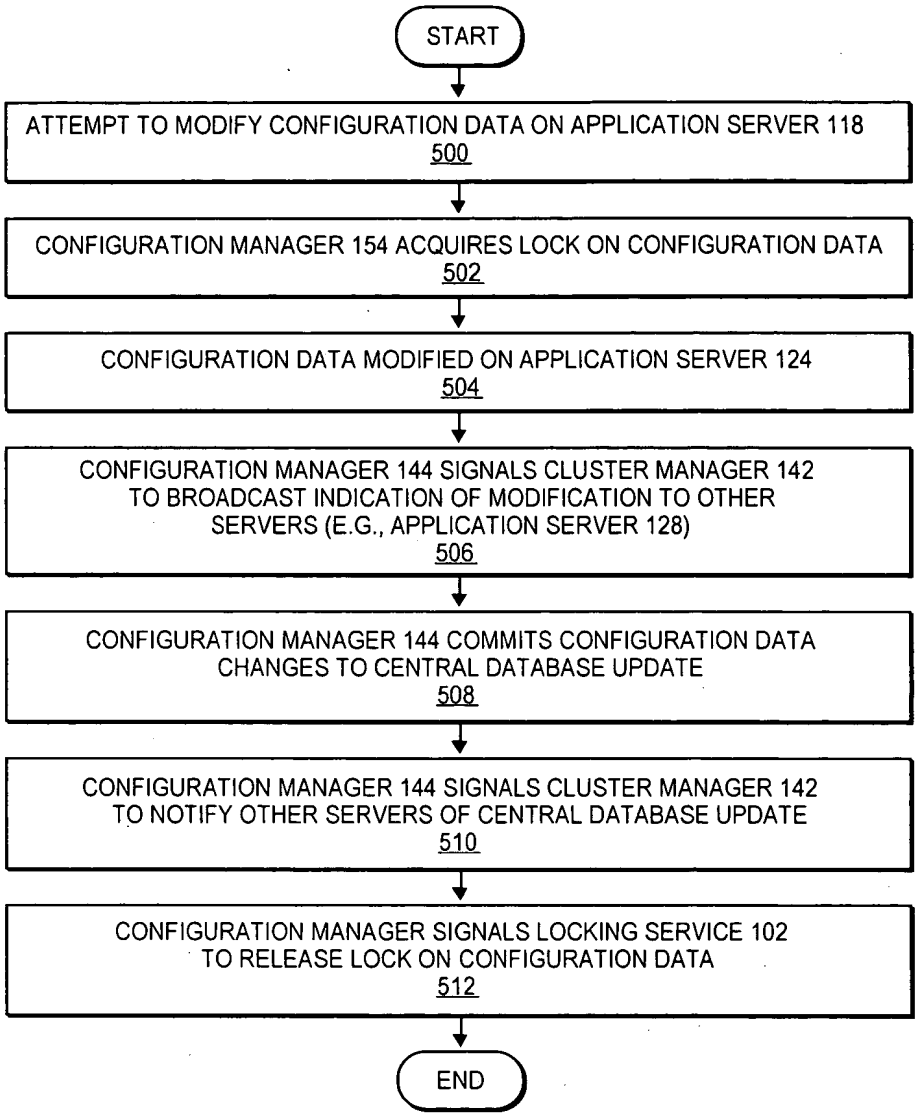
START
ATTEMPT TO MODIFY CONFIGURATION DATA ON APPLICATION SERVER 118
500
CONFIGURATION MANAGER 154 ACQUIRES LOCK ON CONFIGURATION DATA
502
CONFIGURATION DATA MODIFIED ON APPLICATION SERVER 124
504
CONFIGURATION MANAGER 144 SIGNALS CLUSTER MANAGER 142 TO BROADCAST INDICATION OF MODIFICATION TO OTHER SERVERS (E.G., APPLICATION SERVER 128)
506
CONFIGURATION MANAGER 144 COMMITS CONFIGURATION DATA CHANGES TO CENTRAL DATABASE UPDATE
508
CONFIGURATION MANAGER 144 SIGNALS CLUSTER MANAGER 142 TO NOTIFY OTHER SERVERS OF CENTRAL DATABASE UPDATE
510
CONFIGURATION MANAGER SIGNALS LOCKING SERVICE 102 TO RELEASE LOCK ON CONFIGURATION DATA
512
END

FIG. 5A

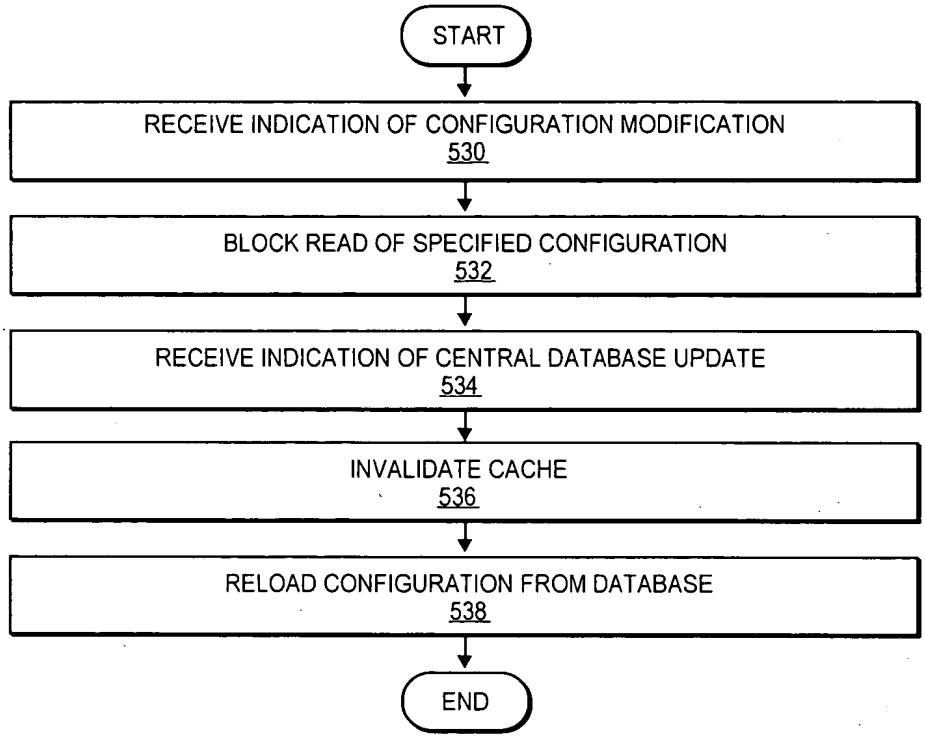
START
RECEIVE INDICATION OF CONFIGURATION MODIFICATION
530
BLOCK READ OF SPECIFIED CONFIGURATION
532
RECEIVE INDICATION OF CENTRAL DATABASE UPDATE
534
INVALIDATE CACHE
536
RELOAD CONFIGURATION FROM DATABASE
538
END

FIG. 5B

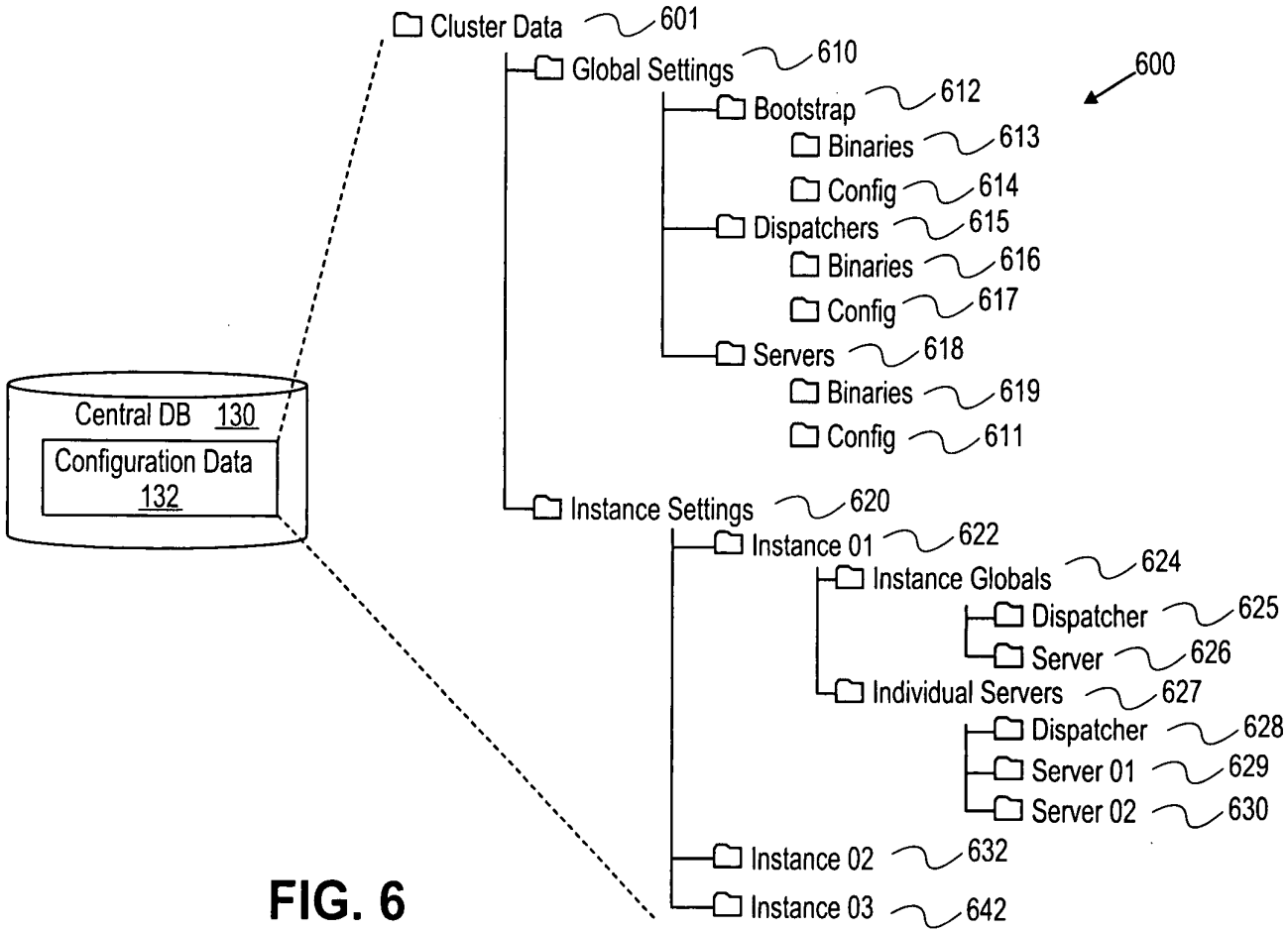
Cluster Data 601
Global Settings 610
Bootstrap 612
Binaries 613
Config 614
Dispatchers 615
Binaries 616
Config 617
Servers 618
Binaries 619
Config 611
600
Central DB 130
Configuration Data 132
Instance Settings 620
Instance 01 622
Instance Globals 624
Dispatcher 625
Server 626
Individual Servers 627
Dispatcher 628
Server 01 629
Server 02 630
Instance 02 632
Instance 03 642

FIG. 6